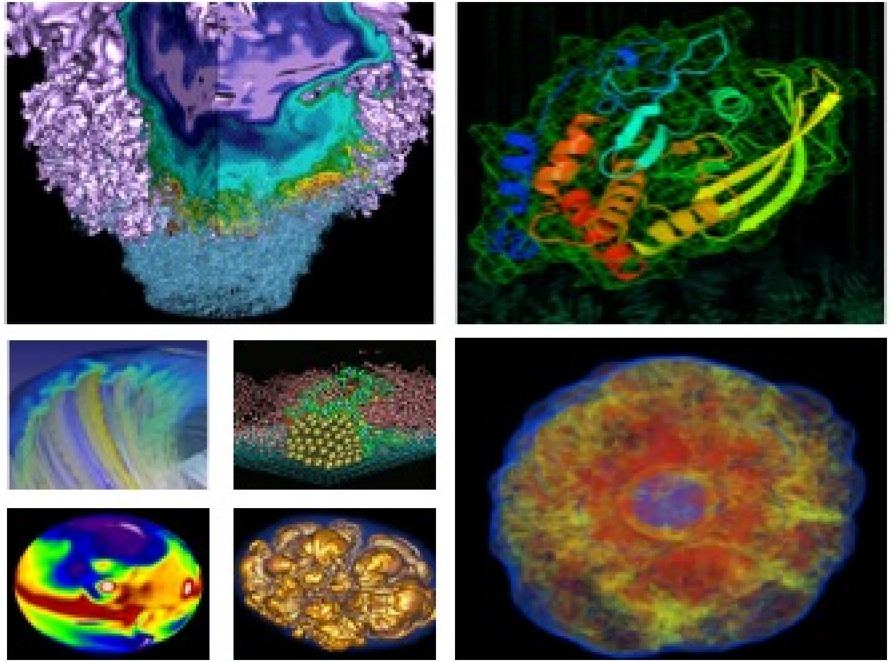
# Deep generative modeling for HEP, Cosmology and Fluid Dynamics

APS Physics Next Workshop
October 8th 2018



Karthik Kashinath
(feat. Prabhat, Wahid Bhimji, Mustafa Mustafa, Debbie Bard, Steve Farrell, Adrian Albert, Ben Nachmann, Michela Paganini and others)
Lawrence Berkeley Lab (LBNL)

NERSC

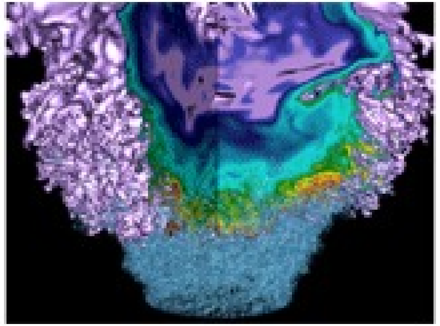
U.S. DEPARTMENT OF ENERGY | Office of Science

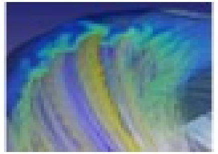
BERKELEY LAB

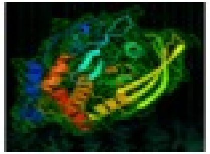
# Outline

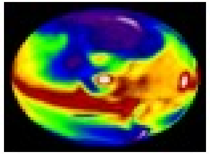
- **Introduction**
  - High Energy Physics (HEP), Cosmology and Fluid Dynamics
  - Deep Learning @ NERSC
- **Current deep generative modeling in HEP, Cosmology and Fluid Dynamics**
  - Cosmology: CosmoGAN
  - HEP: CaloGAN, whole event GAN
  - Fluid Dynamics: physics-constrained GAN
- **Challenges and opportunities**

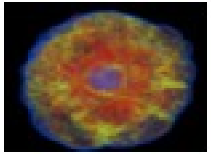
# Introductions

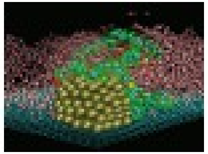




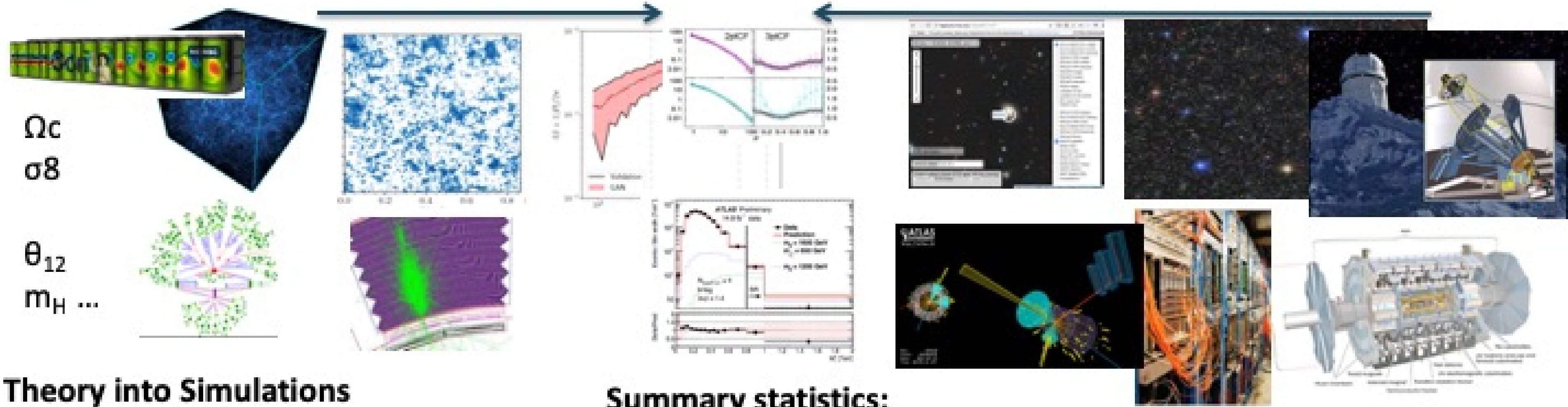








# HEP/Cosmology/Turbulence in practice




**Theory into Simulations**

- Cosmology: high-resolution; produce mass densities; populate with galaxies
- HEP: detailed physics and detector simulation
- Turbulence: high-res simulation

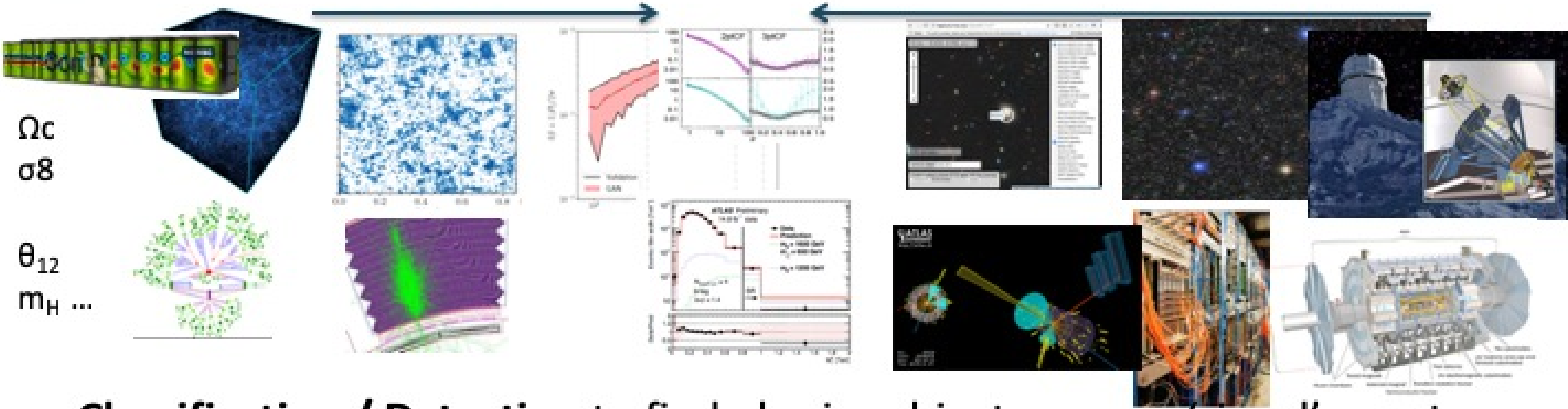
**Summary statistics:**

- E.g. 2pt /3pt correlation: spatial distribution
- E.g. Masses of reconstructed particles
- E.g. Kinetic energy spectra

**Exp/Obs reconstruction**

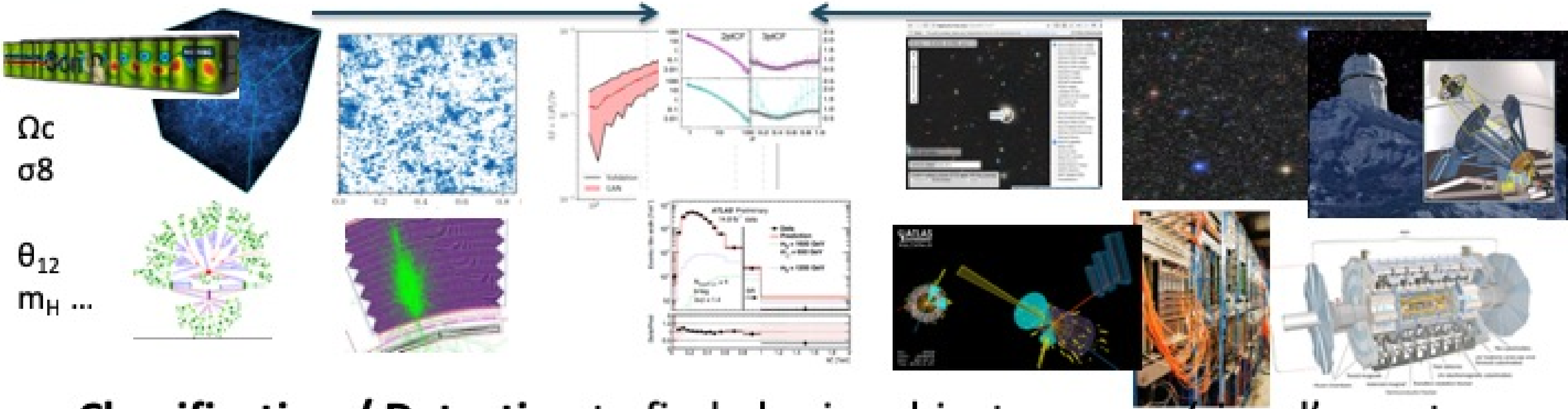
- Derive position of galaxies/stars and properties for catalogs
- Reconstruct particle properties
- Detect coherent spatio-temporal patterns

# How can Machine Learning help?



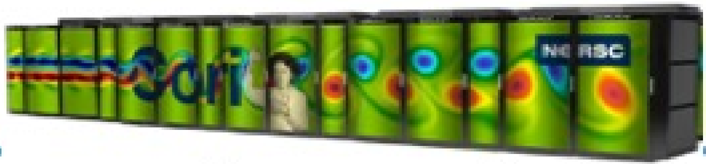

- **Classification / Detection** to find physics objects or new 'signal' events or coherent structures
- **Regression** to aid reconstruction of fundamental physics parameters
- **Clustering / feature detection** in high-dimensional raw data
- **Generation** of data to augment expensive hi-res hi-fi simulations

# How can Machine Learning help?

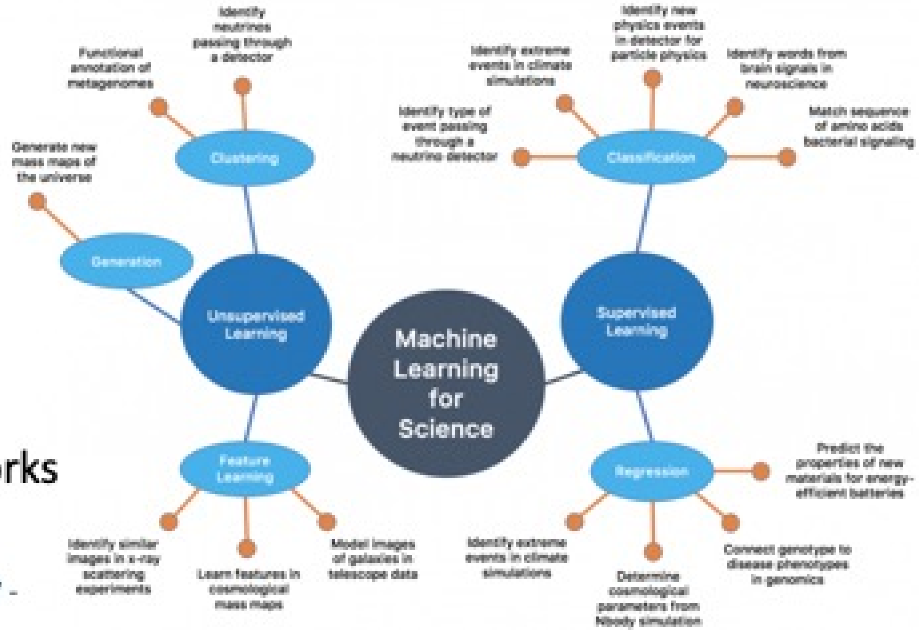



- **Classification / Detection** to find physics objects or new 'signal' events or coherent structures
- **Regression** to aid reconstruction of fundamental physics parameters
- **Clustering / feature detection** in high-dimensional raw data
- **Generation of data to augment expensive hi-res hi-fi simulations**

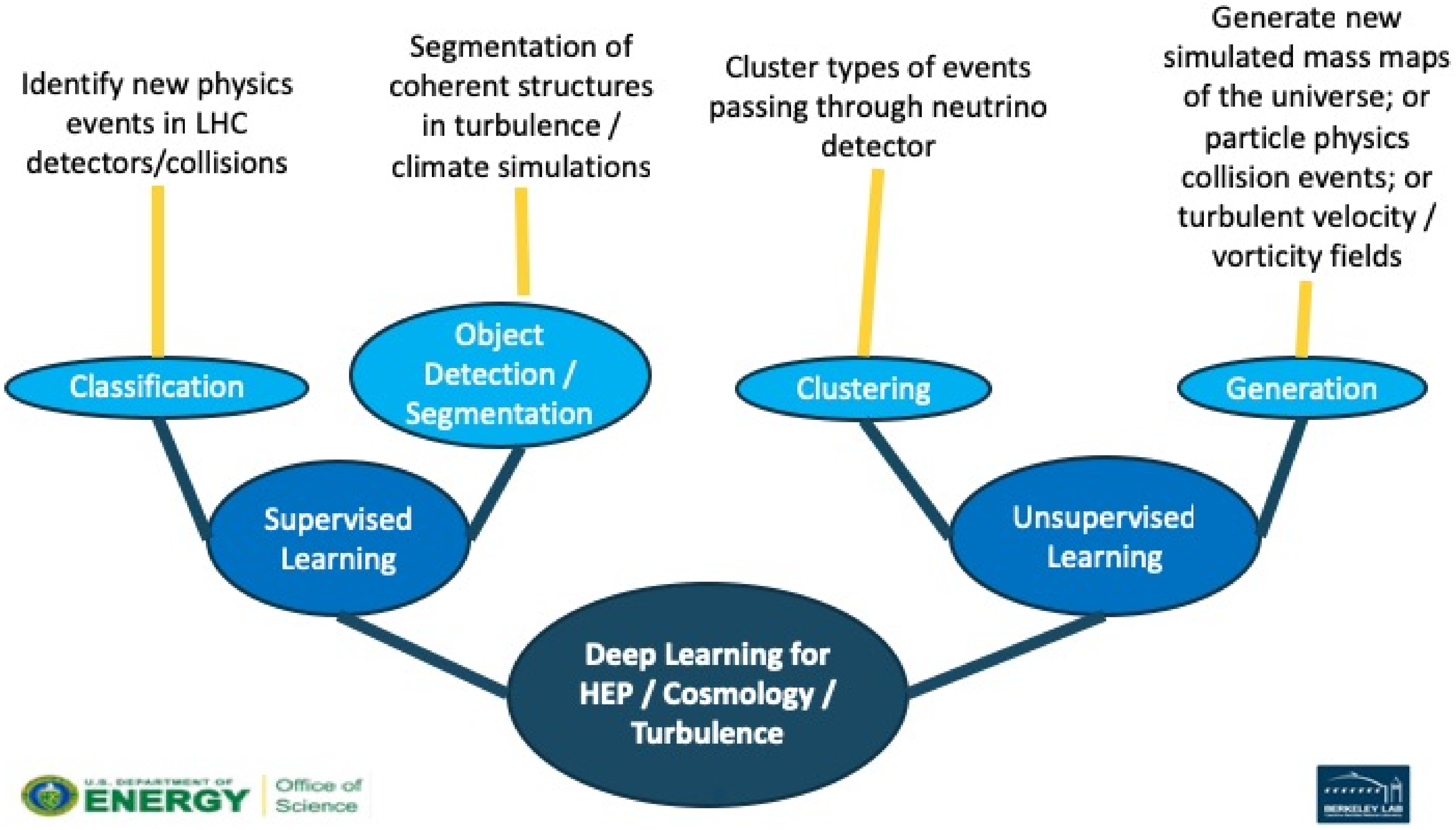
# NERSC




- **Mission high-performance computing center for US Dept. of Energy**
  - >7000 users across many science areas (e.g. cosmology, climate science, bioscience)
- **Cori – NERSC's Newest Supercomputer – Cray XC40 (31.4 PF Peak)**
  - 9668 Intel Knights Landing nodes
- **Data and Analytics (DAS) group:**
  - Develop cutting-edge methods;
  - Build tools for ML / DL;
  - Optimize those for large scales;
  - Collaborate on Projects / Training
  - Support (most) deep learning frameworks

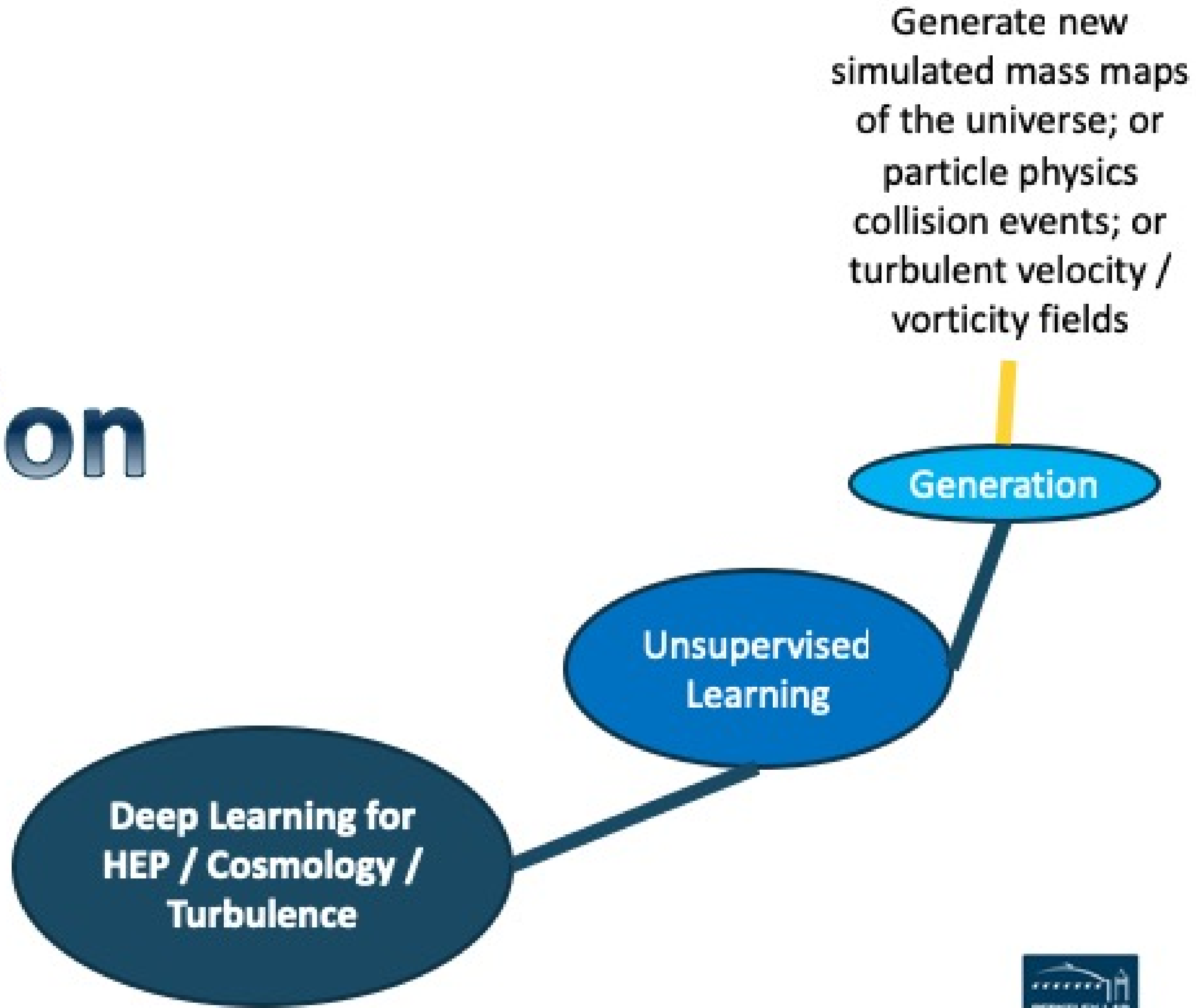



U.S. DEPARTMENT OF ENERGY | Office of Science



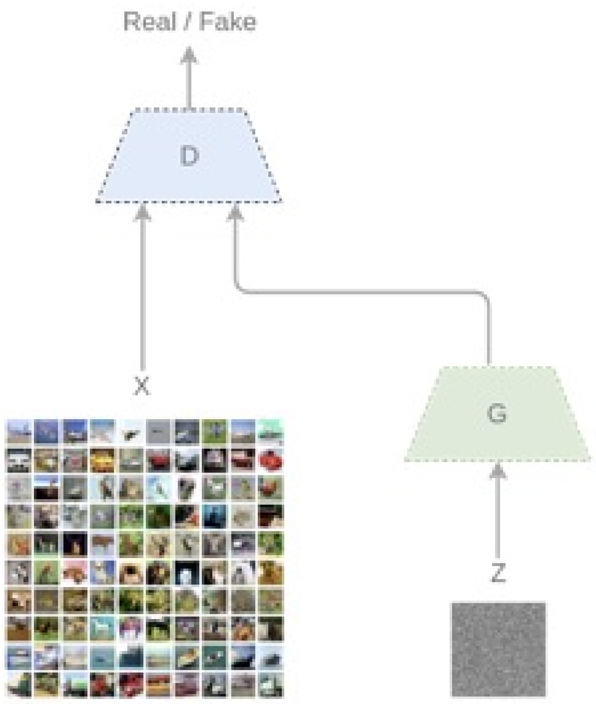
Identify new physics events in LHC detectors/collisions
Segmentation of coherent structures in turbulence / climate simulations
Cluster types of events passing through neutrino detector
Generate new simulated mass maps of the universe; or particle physics collision events; or turbulent velocity / vorticity fields
Classification
Object Detection / Segmentation
Clustering
Generation
Supervised Learning
Unsupervised Learning
Deep Learning for HEP / Cosmology / Turbulence
U.S. DEPARTMENT OF ENERGY
Office of Science
BERKELEY LAB

# Generation


Generate new simulated mass maps of the universe; or particle physics collision events; or turbulent velocity / vorticity fields
Generation
Unsupervised Learning
Deep Learning for HEP / Cosmology / Turbulence

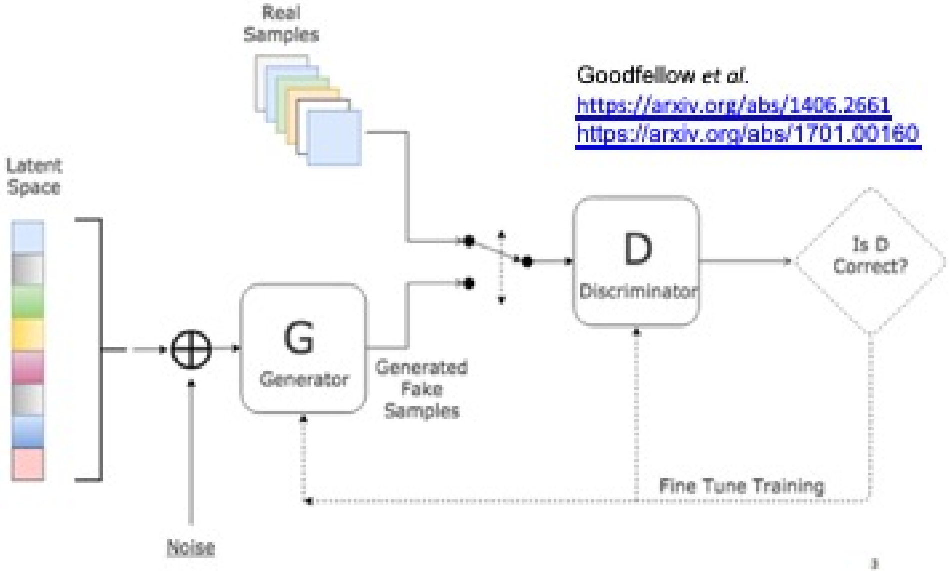
# Deep learning generative models

- **Deep neural networks that learn to sample from a data distribution**
  - Transform a simple “noise” distribution to the target distribution
- **Popular examples:**
  - Variational Autoencoder (VAE)
  - Generative Adversarial Network (GAN)
- **The GAN framework poses the problem as a trainable two-player game**
  - A *generator* tries to produce realistic samples
  - A *discriminator* tries to distinguish real from fake samples



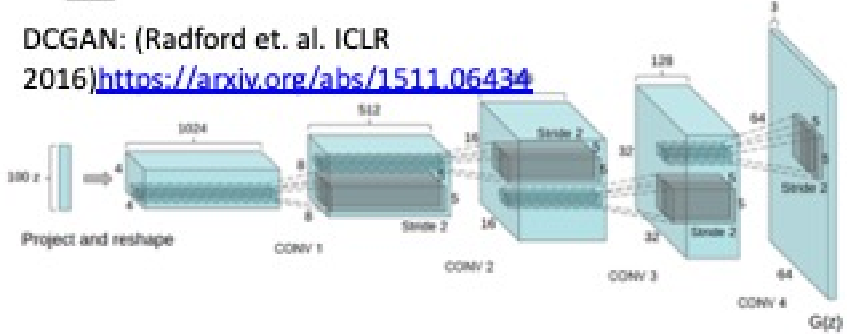
# Generative Adversarial Networks (GAN)

- Jointly optimize Discriminator (D) and Generator (G) NNs
  - G architecture like decoder in VAE
  - Loss for G/D in opposition




Goodfellow *et al.*
https://arxiv.org/abs/1406.2661
https://arxiv.org/abs/1701.00160

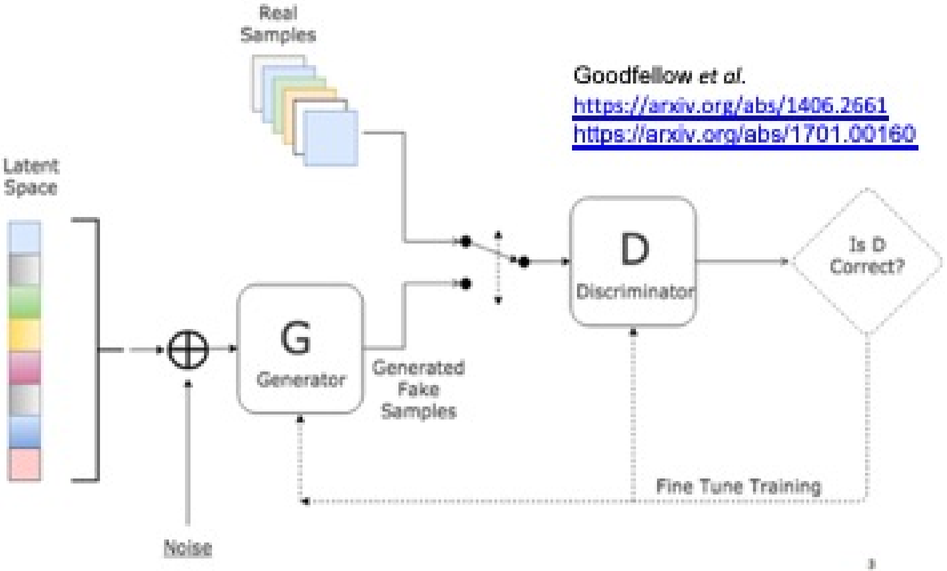
DCGAN: (Radford et. al. ICLR 2016)https://arxiv.org/abs/1511.06434



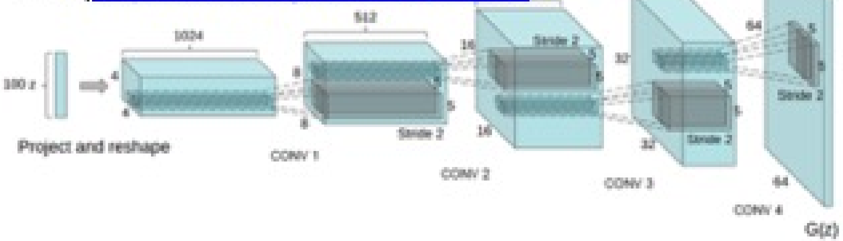
# Generative Adversarial Networks (GAN)

- Jointly optimize Discriminator (D) and Generator (G) NNs
  - G architecture like decoder in VAE
  - Loss for G/D in opposition
- On ‘natural images’ GANs can be unstable, our problems have advantages:
  - underlying physics structure
  - existing simulation samples
  - metrics to evaluate
- Build on industry research – e.g. convolutional DCGAN

Goodfellow *et al.*
https://arxiv.org/abs/1406.2661
https://arxiv.org/abs/1701.00160

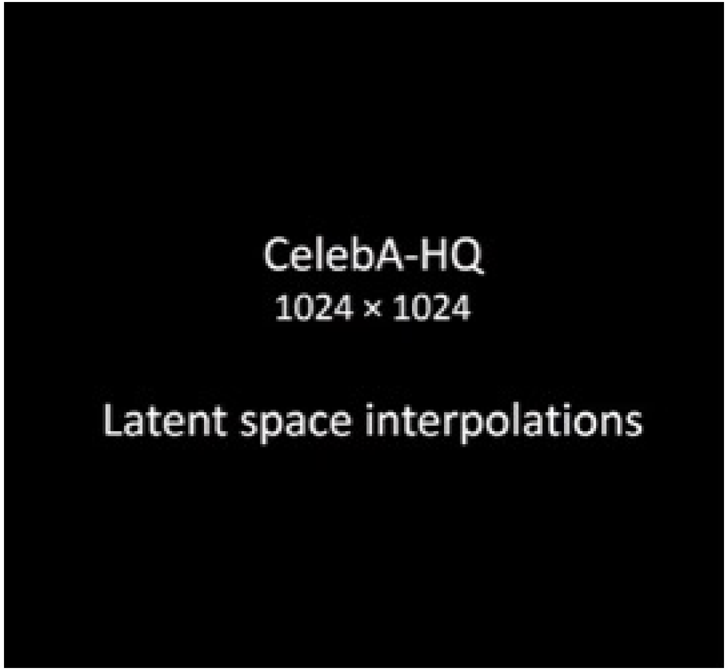



DCGAN: (Radford et. al. ICLR 2016)https://arxiv.org/abs/1511.06434

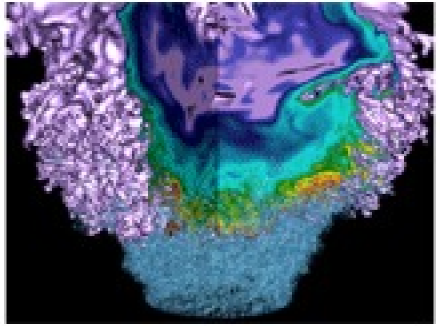


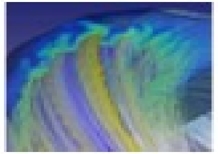
# Science Use Case 1: Celebrities!

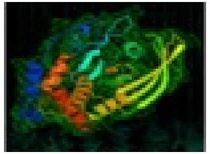
[Karras et. al. ICLR 2018]

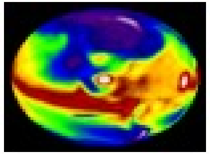


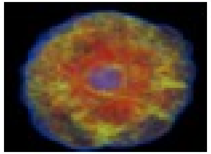
# Science Use-cases

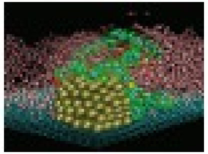




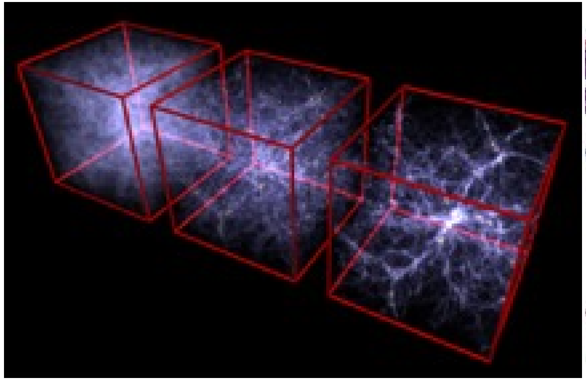


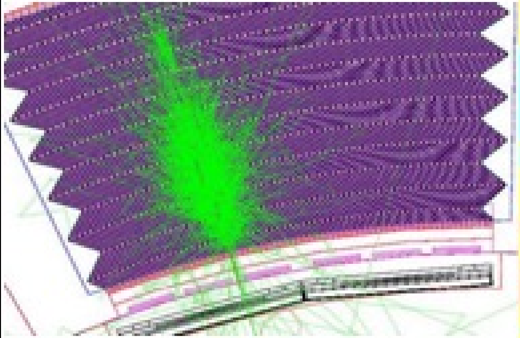


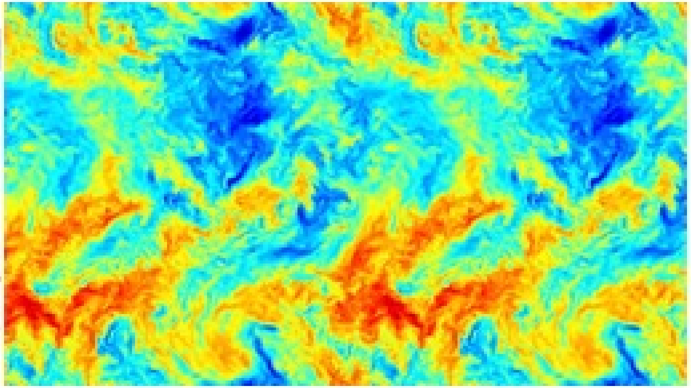


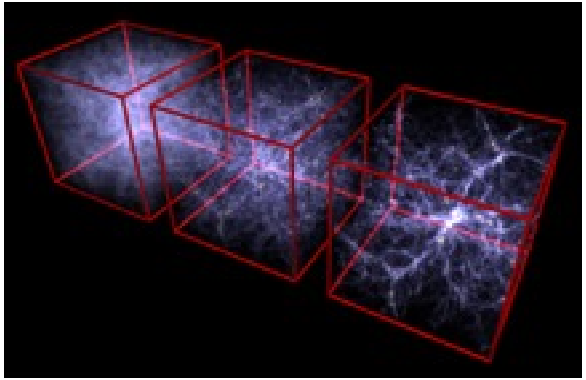


1. Simulations essential to science
2. Often computationally expensive
3. Sometimes no good simulation exists







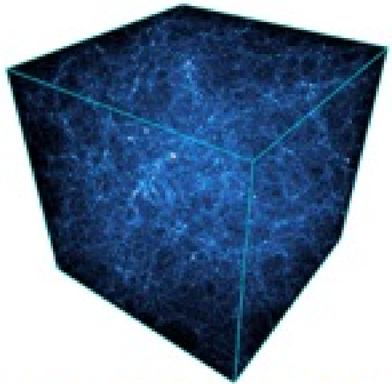


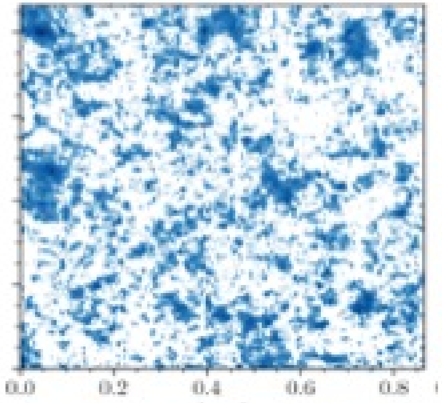
# Cosmology

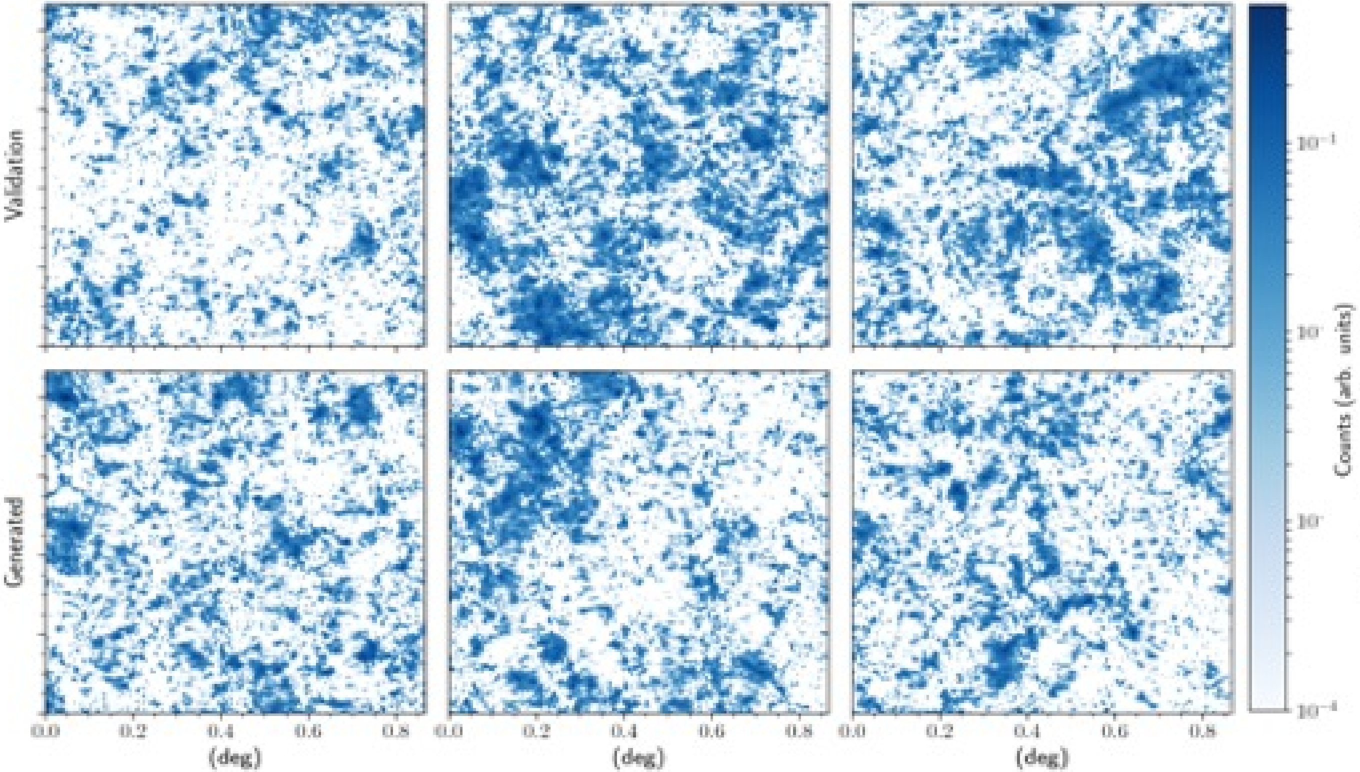
# CosmoGAN

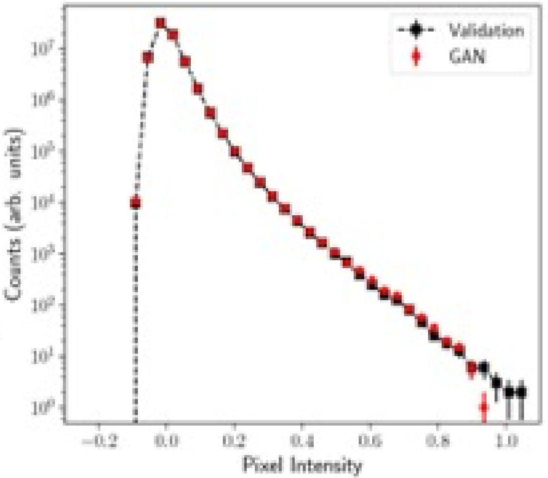
Mustafa Mustafa, Deborah Bard, Wahid Bhimji, Rami Al-Rfou, Zarija Lukić
https://arxiv.org/abs/1706.02390

- **Cosmology simulations create 'virtual universes'**
  - 'Q-continuum' HACC sim took ~ 2 weeks on a Cori-sized HPC resource
  - One product is weak lensing convergence maps, to compare to observations
- **Seek to augment simulations with a learned generative neural net**
  - Reproduce convergence maps– images
  - Train using existing simulation maps
  - Once trained produce new examples
- **Use DCGAN type architecture on 256x256 images**



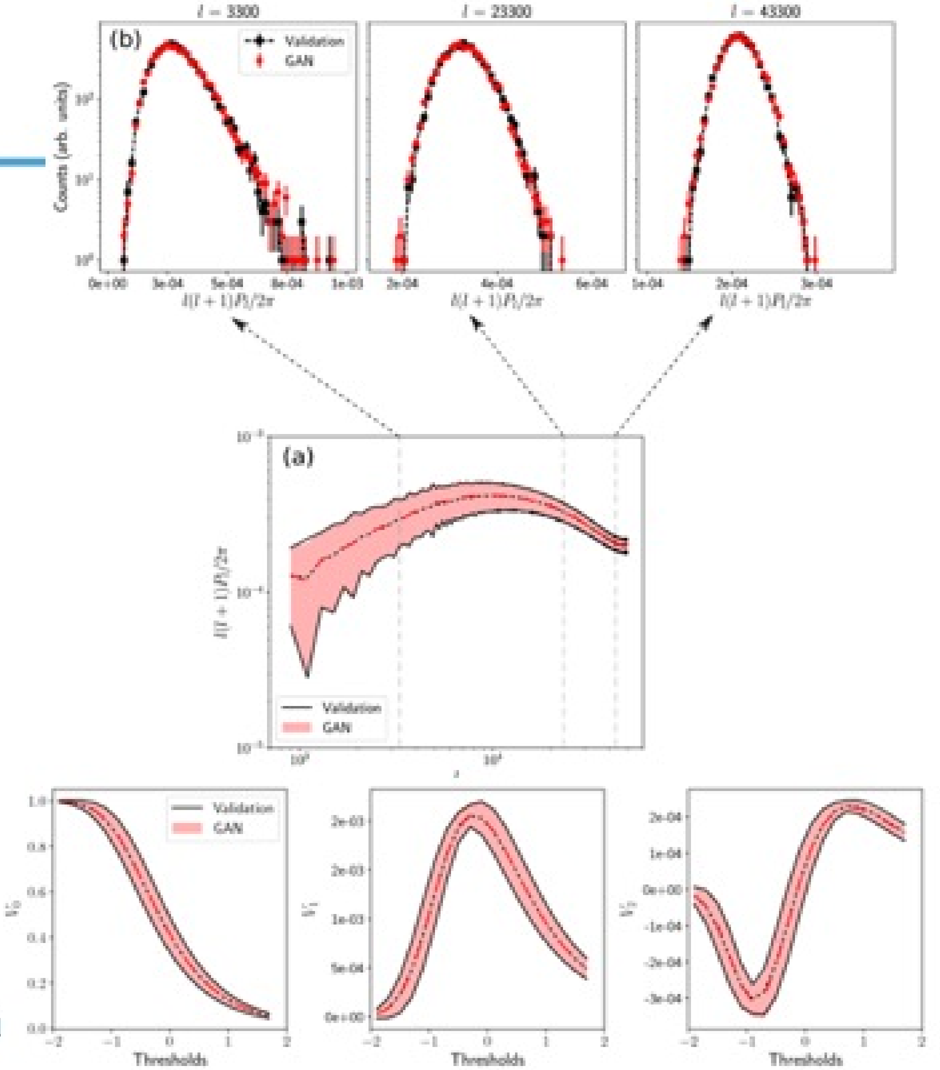



Validation
Generated
(deg)
(deg)
(deg)


Validation
GAN
Counts (arb. units)
Pixel Intensity

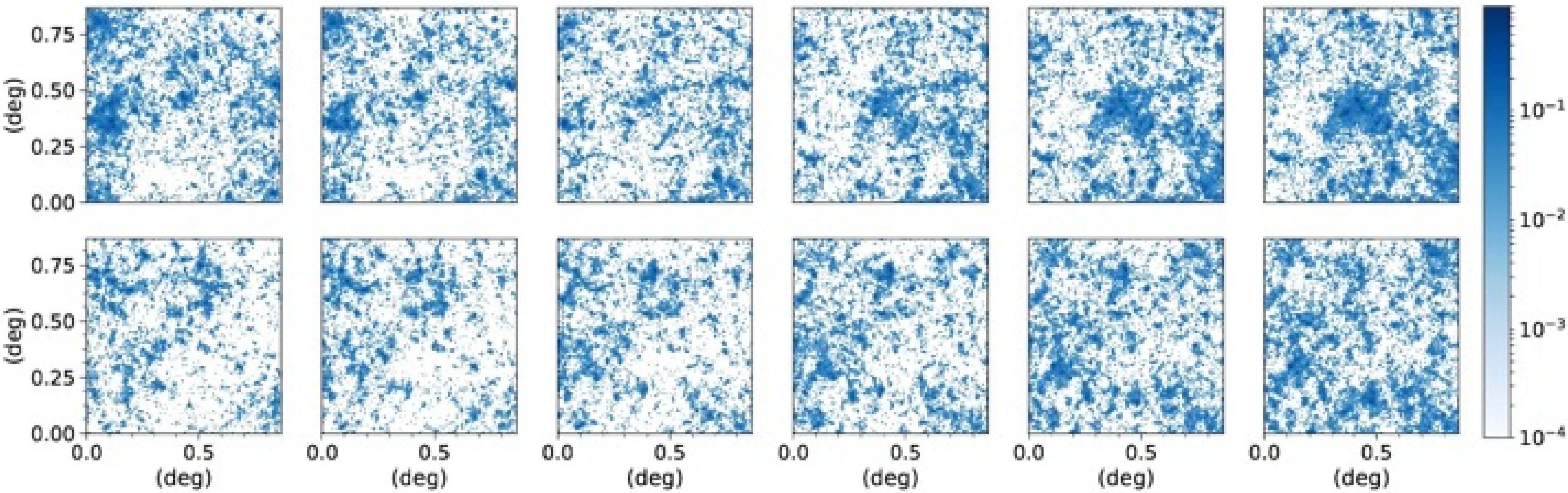
# Summary statistics

- **Calculate power spectrum for generated images and validation sample**
  - Fourier transform of 2pt correlation
  - Excellent agreement (K-S p_value > 0.995 for 246/248 moments)
  - GAN not explicitly trained to reproduce these distributions
  - Also higher-order *Minkowski functionals* are reproduced

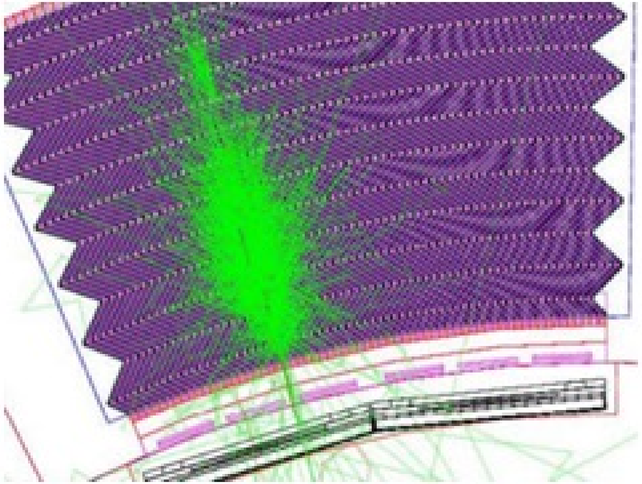


# Interpolation

- **Varies smoothly between two random input vectors**



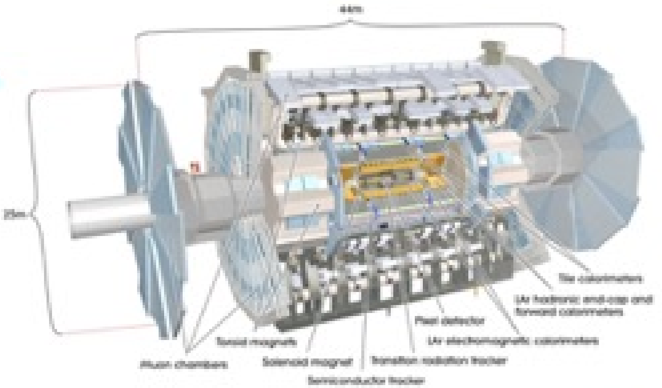
# Particle Physics

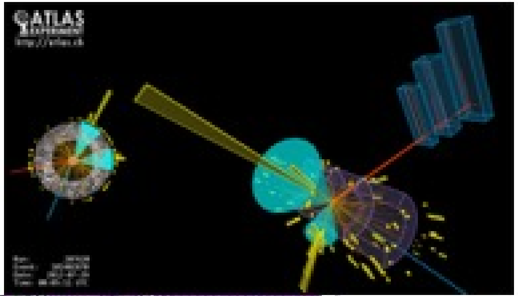


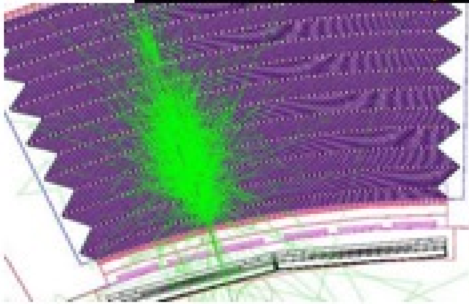
Paganini, de Oliveira, Nachmann [Phys. Rev. Lett. 120, 042003]

# CaloGAN

- **Collisions in detectors e.g. at the Large Hadron Collider (LHC)**
- **Need detailed micro-physics detector simulations to model**
  - > ~ 50% LHC computing budget (> $10^9$ CPU hours in 2017)
  - Much of this compute time in a 'shower' in part of the detector
  - Faster simulation approaches not precise enough for many applications

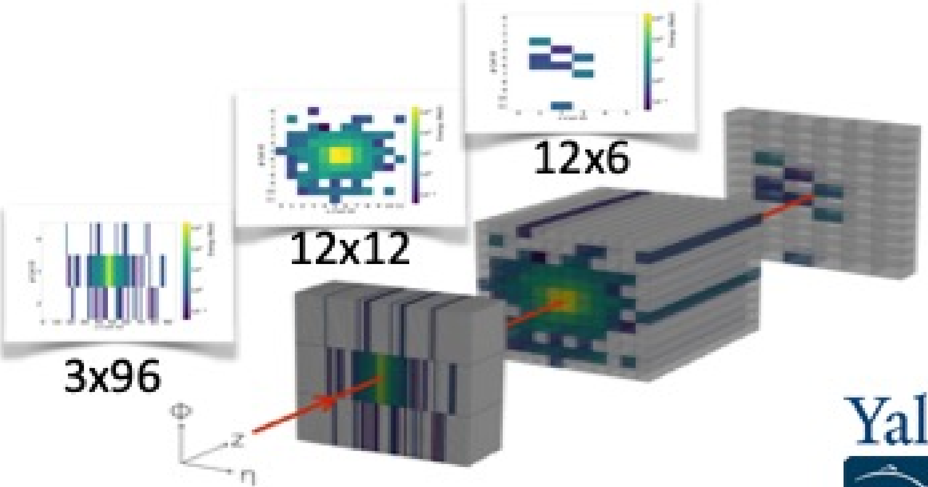




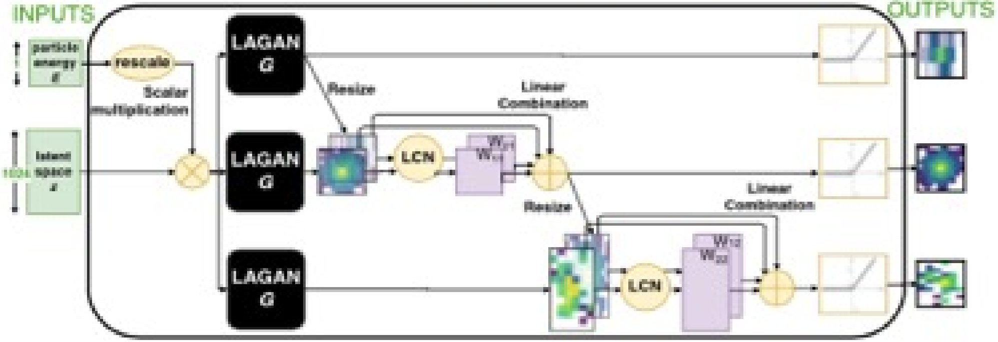



# CaloGAN

- **CaloGAN models a 3-layer calorimeter like that of the ATLAS LHC experiment**
- **Custom NN design**
  - Sparsity
  - High dynamic range
  - Highly location-dependent features
- ***Conditioned* with particle energy as input**

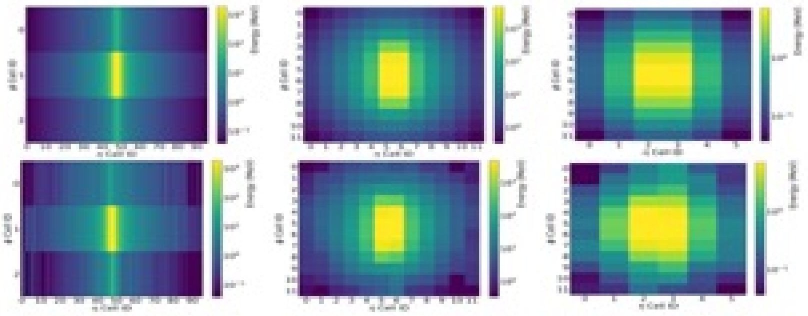



Paganini, de Oliveira, Nachmann [Phys. Rev. Lett. 120, 042003](https://doi.org/10.1103/PhysRevLett.120.042003)



# CaloGAN - results

- **Realistic averages and individual images**
- **Conditional generation based on particle energy**
  - Allowing parameter interpolation
- **Fast**
  - 5M particle showers per min on GPU while full simulation on CPU is 30 showers/min



Average energy deposition per calorimeter layer in the GEANT4 training dataset (top) and in the GAN generated dataset (bottom)

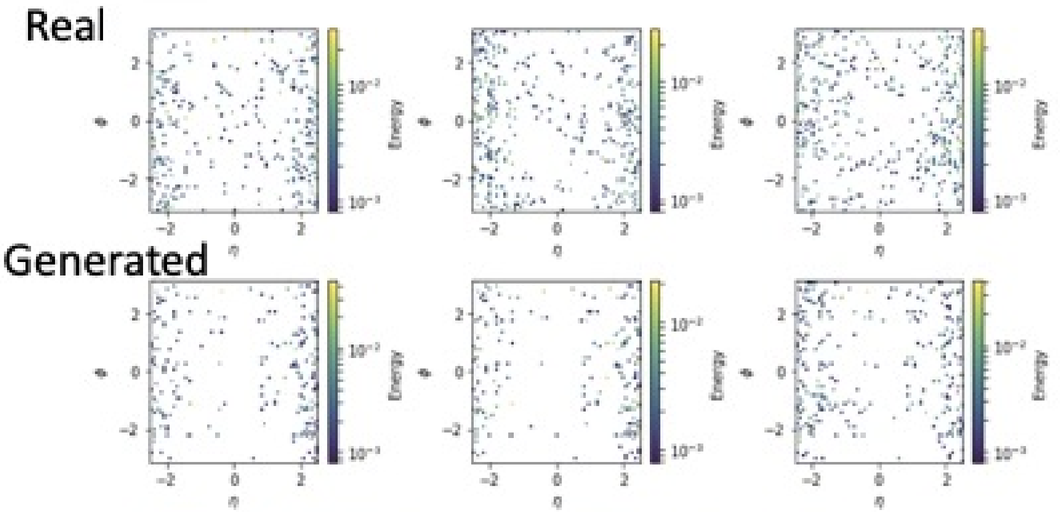

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Requested (GeV) | 1.0 | 23.1 | 45.2 | 67.3 | 89.4 | 111.6 | 133.7 | 155.8 | 177.9 | 200.0 |
| Generated (GeV) | 1.4 | 23.1 | 46.8 | 69.3 | 91.6 | 113.6 | 135.2 | 159.3 | 184.4 | 211.2 |

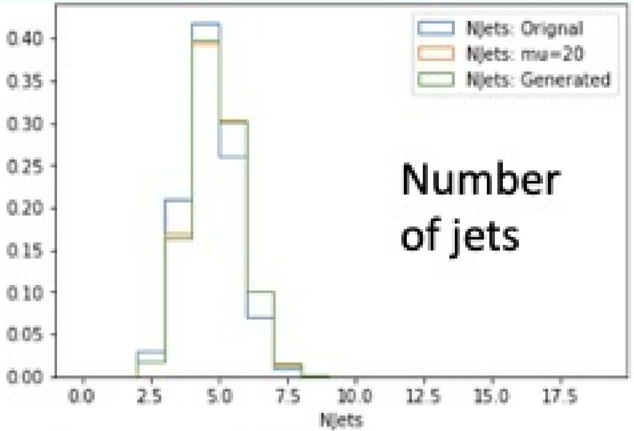
Paganini, de Oliveira, Nachmann Phys. Rev. Lett. 120, 042003

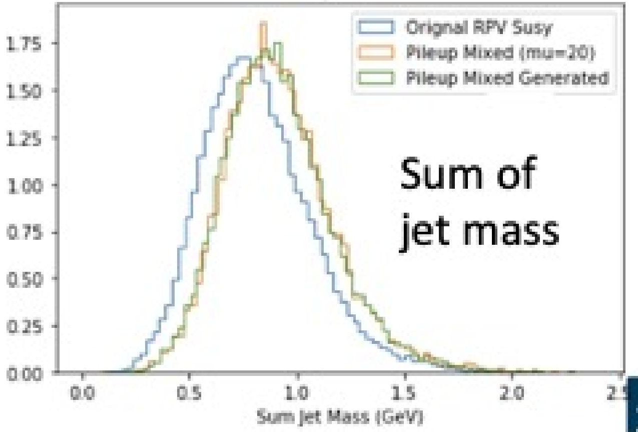
# Pileup GAN

- **Pileup poses big challenges for HL-LHC computing workflows**
  - Need to simulate and store a very large volume of pileup events
  - Need to read this data from disk and overlay during digitization
- **The distribution can be modeled with a whole-detector GAN**
  - Simulate samples and train model *once*
  - Use the trained generator for fast, on-the-fly pileup sampling
- **To test fidelity, now we can evaluate the effects on reconstructed object kinematics**
  - Overlay real pileup or generated pileup
  - Compare the shifts in the distributions

# Pileup GAN - μ=20




- **The GAN gives realistic looking pileup images**
- **When overlaid onto RPV events, we see realistic shifts in the distributions**

# Whole detector GAN

**Can a GAN be trained to learn the distribution of full detector images?**

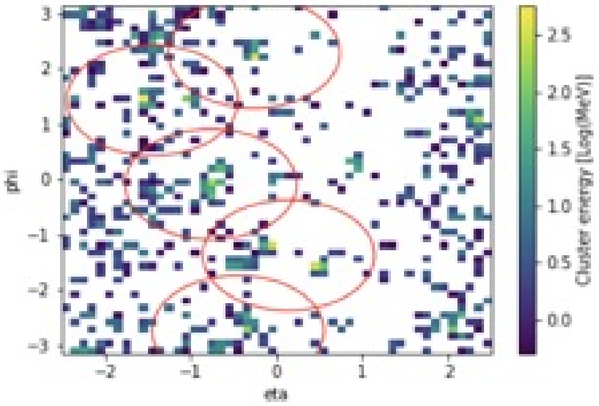
- In HEP, previously only applied to individual *particles*

**Can the generator learn to produce realistic jets?**

- Reconstruct-able, with the correct distributions?

**How could we use such a model?**

- Theory parameter interpolation
- Pileup simulation
- Other possibilities



# Whole detector GAN

### Data

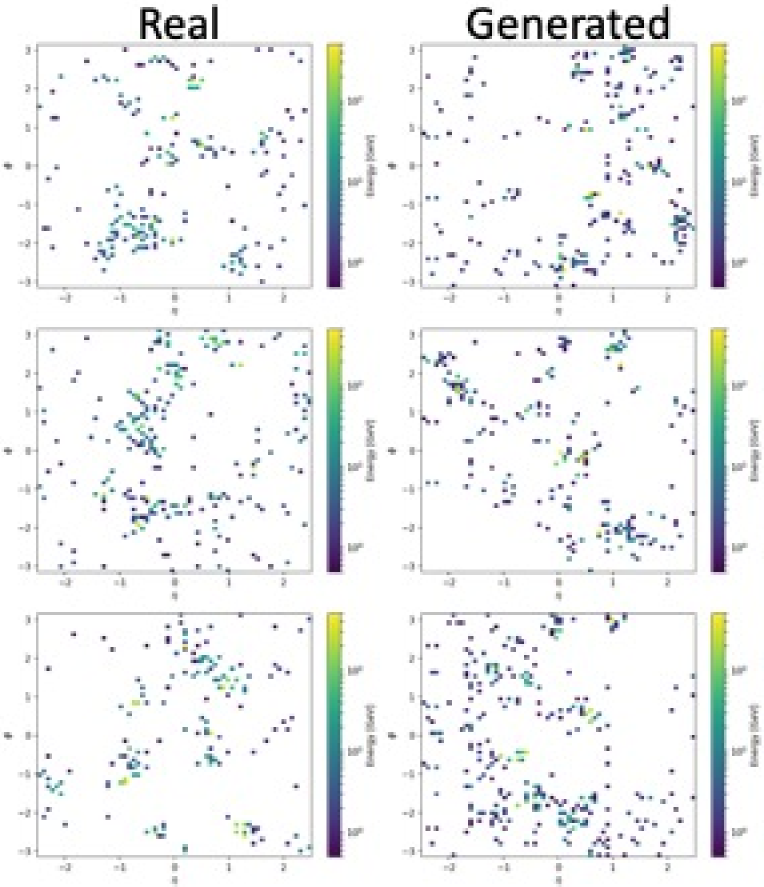
- 64 x 64 x 1 images representing calorimeter tower energy
- Delphes+Pythia simulation using ATLAS detector card

### Architecture

- "Standard" DCGAN topology in generator and discriminator
- Threshold on the generator output for sparsity

### Evaluation

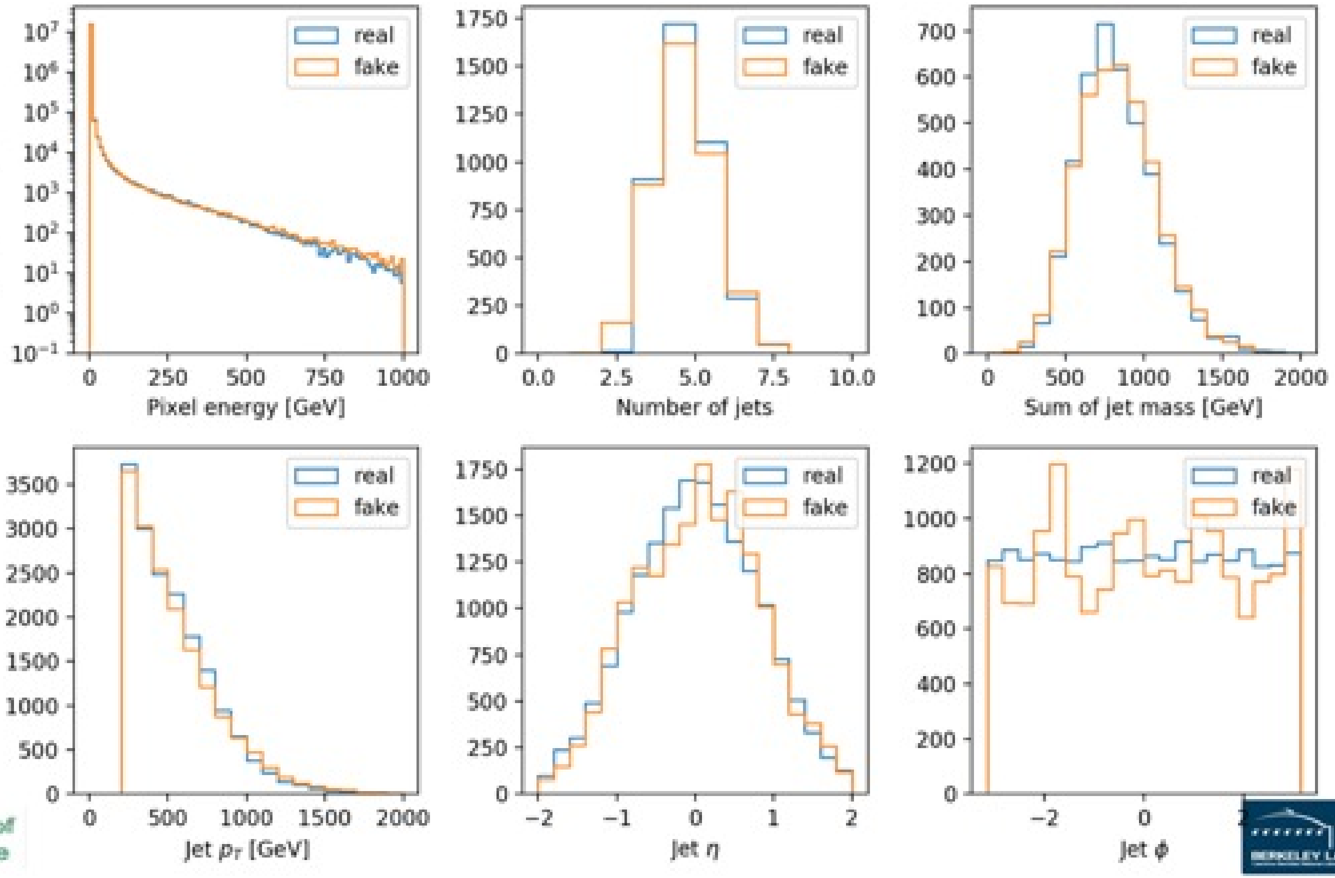
- Images reconstructed with FastJet (R=1, pt>200GeV)
- *Kolmogorov-Smirnoff* test used to compare real and generated jet distributions (#, momenta, directions, etc.)
- KS metric used to select best model and epoch in *random hyper-parameter search*



# Whole detector GAN

**The generated samples produce realistic jet multiplicities and kinematics**

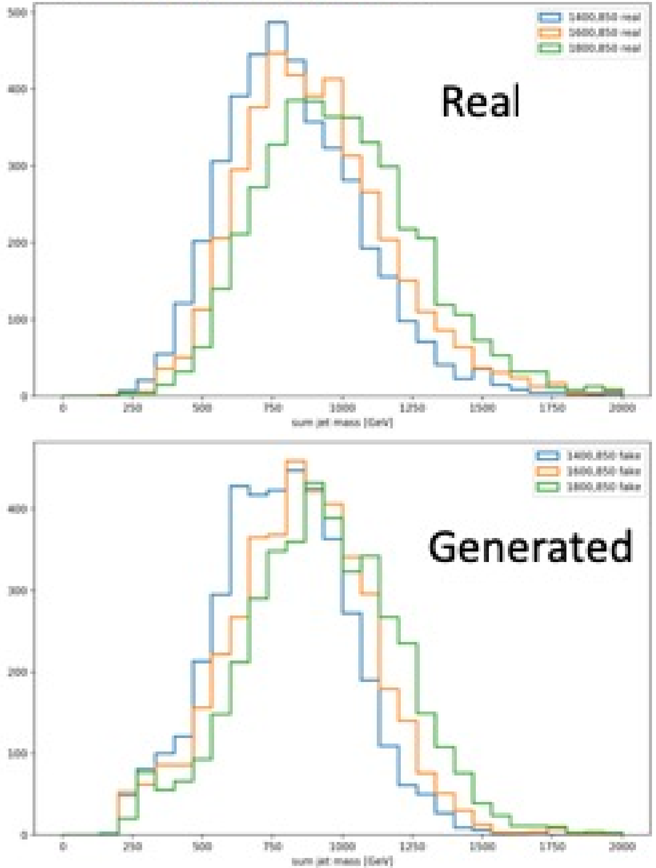
**This is without imposing *any* physics knowledge!**



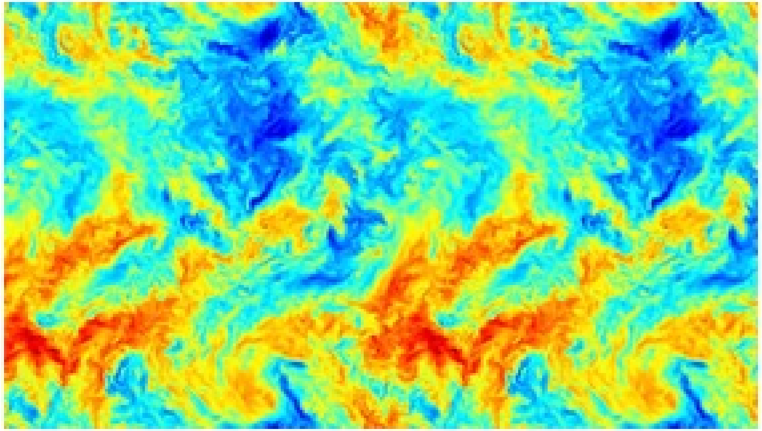
# Conditional Whole Detector GAN

- **Can the GAN learn to produce images conditional on the SUSY theory parameters?**
  - We augment the discriminator and generator to be conditioned on $M^{glu}$,$M^{neu}$
- **The GAN is shown to learn the conditional distributions**
  - E.g., summed jet mass shifts as expected
- **Could use this to supplement full simulation in MC signal grids**
  - Coarse full-sim grid
  - Interpolate with GAN

$M_{glu}$ = [1400, 1600, 1800] GeV



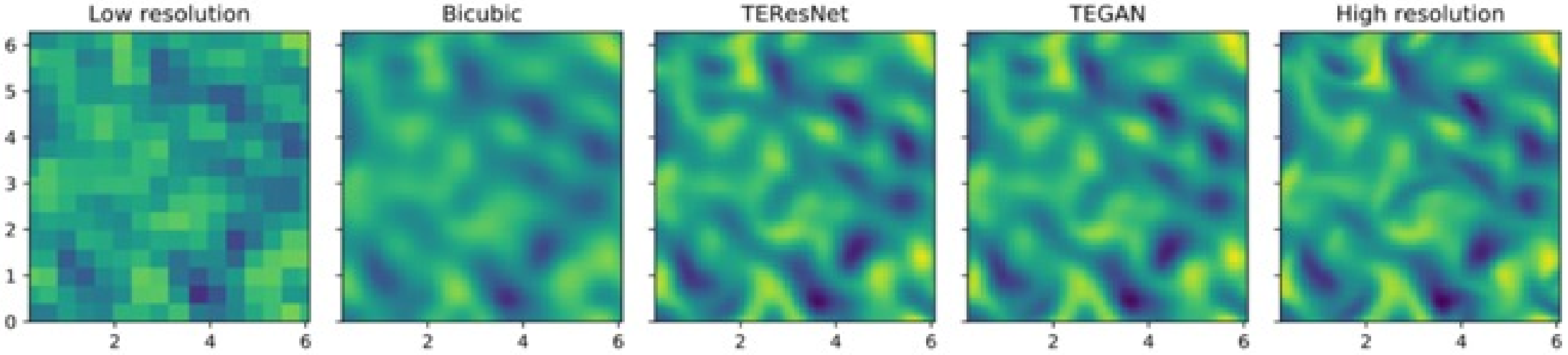

Work with Ben Nachmann (LBNL), Harley Patton (Berkeley)



# Turbulence

# Physics-constrained GAN

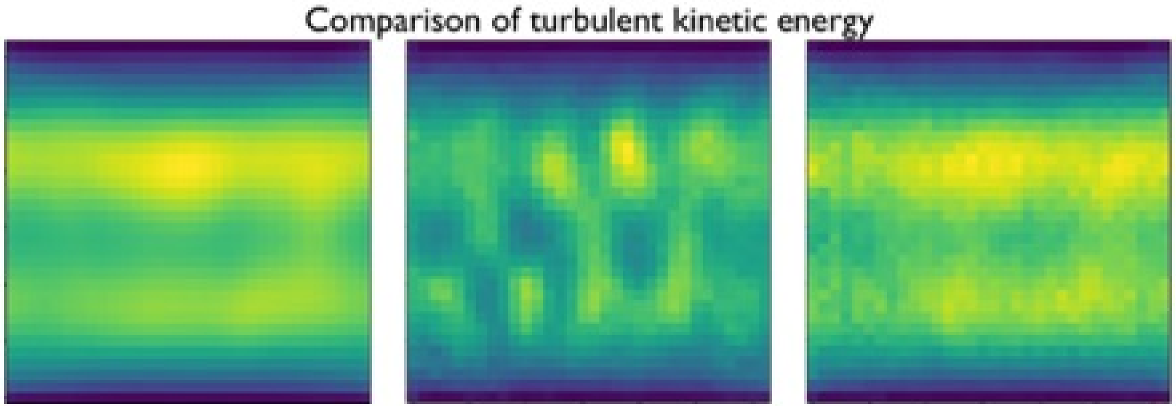
- Turbulence simulations need reliable sub-grid scale models
- Can we impose constraints for turbulence super-resolution? (physical or statistical)

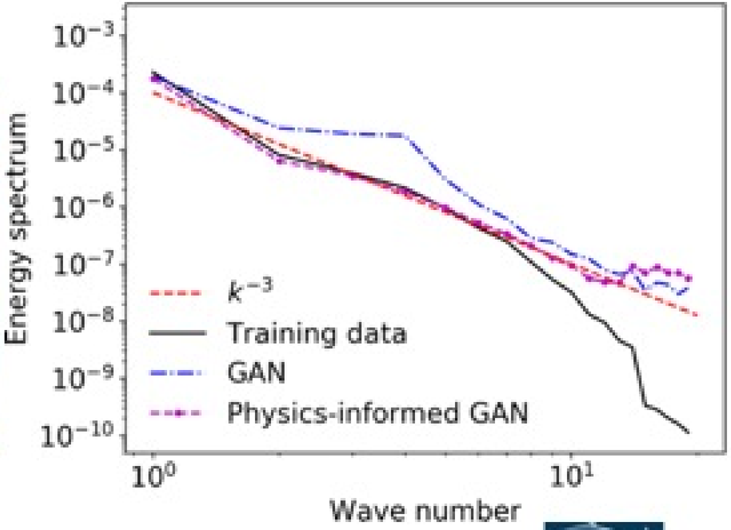



Raunak Borker, Sravya Nimmagadda, Akshay Subramaniam and Man-Long Wong, 2018

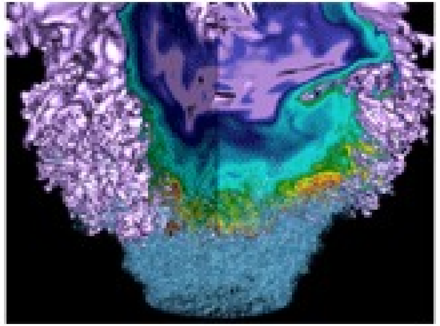
# Physics-constrained GAN

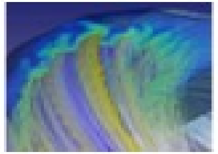
- **Constrained GAN converges up to 10x faster than unconstrained GAN**
- **Constrained GANs seem to be more stable**

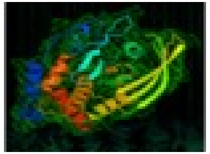


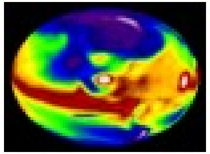



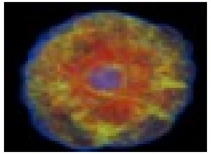
# Future opportunities and challenges

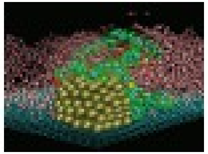






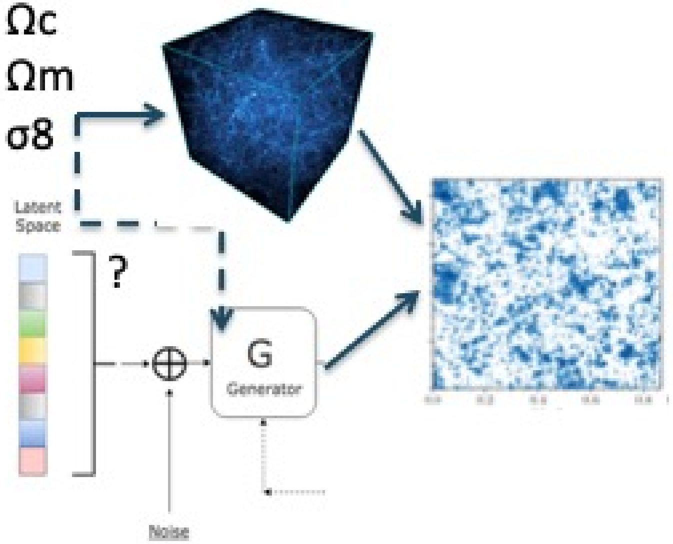






# Opportunities

- **GANs can reproduce physical features and learn statistical properties**
  - Including non-Gaussian structures and derived variables
  - With high-accuracy in many cases
  - And fast
- **Many promising opportunities**
  - Reliably interpolate in parameters for fast emulator
  - Compress simulations to model and generate on the fly
  - Create simulations from data *without* a physics model



# AND Challenges

**Computational**

- Significant compute required for training
- Challenges in distributed training on HPC resources

**Methodological**

- Issues in stability and convergence, especially at scale
- Can they generalize to arbitrary geometries (beyond images)?
- How to extrapolate in *physics* parameters?
- How to incorporate known physical laws?

**Active/important/impactful area of research – to augment not throw out your HPC simulation**

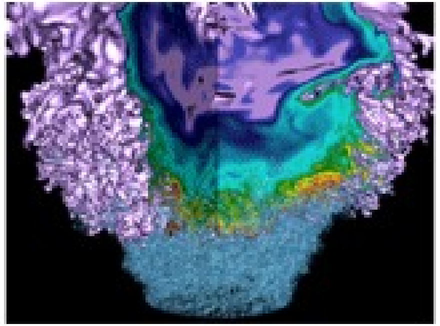
# Conclusions

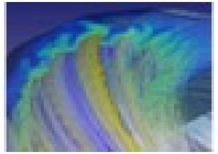
- **Deep generative models show promise for augmenting and accelerating physics simulations as reliable fast emulators**
- **We've demonstrated that GANs can learn complex physics**
  - Can be conditioned on physical parameters (interpolates well)
  - Can incorporate physical / statistical constraints
- **Some applications in 'production' -- more coming…**
  - Potential for huge gains in inference and simulation
- **A number of open challenges remain to bring high-fidelity generative models to practice**
  - Addressing stability, generalizing geometry, etc.

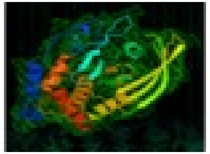
**Questions?**

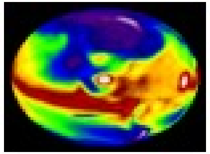
kkashinath@lbl.gov

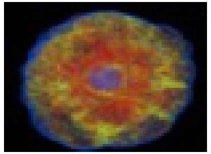
# Backups

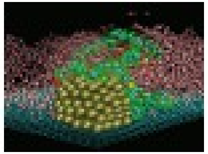












# Generative Adversarial Networks – Loss function

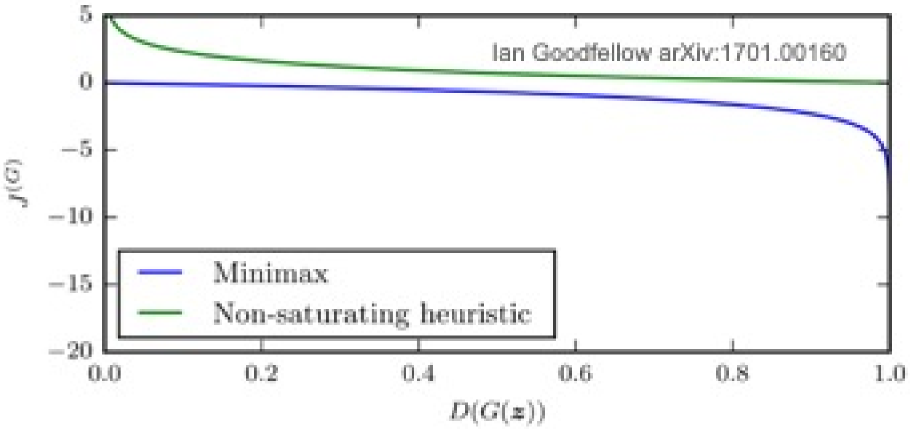
Minimax game formulation (saturating):

$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{x\sim\mathbb{P}_{data}} \log D(x) - \frac{1}{2}\mathbb{E}_{z\sim p_z} \log(1 - D((G(z)))$$

$$J^{(G)} = -J^{(D)}$$

Heuristic loss function (non-saturating):

$$J^{(G)} = -\frac{1}{2}\mathbb{E}_{z\sim p_z} \log D(G(z))$$



# GAN not memorizing training images